

1. Code

First, `tri_dup_mode` is a Boolean that is used to control which retransmission mode to use. By setting it to True, fast retransmission with 3 duplicate acks is chosen. Setting it to false will make the program rely only on timeouts.

`lost_pkt_td` is a list to store retransmitted packet numbers when in fast retransmit mode. `lost_pkt_to` is a list to store retransmitted packet numbers when in timeout only mode. The length of list is printed at the end of program to show results. `latency_sum` is a global variable used to keep track of latency throughout the program. This variable is used with `no_pkt` at the end of program to compute latency and throughput.

The `prev_ack` variable inside the thread target function is used to check whether the received ack is a duplicate ack. If it is, global variable `dup_count` is incremented which is a variable to keep track of number of duplicate acks. When `dup_count` exceeds 3, the program will print '3 dup acks detected', appends packet number to list and retransmits using `tdupack_flag`. `latency_sum` is updated on every reception.

In the main function, a condition is added for retransmitting when 3 duplicate acks are detected. This branch will only execute when `tri_dup_mode` is True.

2. 3 Duplicate Acks vs Timeout

Below is a comparison of 3 duplicate acks and timeout with 1000 total packets, and loss rates of 1%, 5%, 10%, 15% and 20%.

fast retransmit mode(3 dup acks)	fast retransmit mode(3 dup acks)	fast retransmit mode(3 dup acks)	fast retransmit mode(3 dup acks)	fast retransmit mode(3 dup acks)
3 dup loss: 9 loss rate: 0.01	3 dup loss: 49 loss rate: 0.05	3 dup loss: 96 loss rate: 0.1	3 dup loss: 170 loss rate: 0.15	3 dup loss: 199 loss rate: 0.2
latency: 0.000885134696064492	latency: 0.004865984439849854	latency: 0.00890127289848694	latency: 0.018914678573608398	latency: 0.024016147613525392
throughput: 1129.771551645188	throughput: 285.50826094110081	throughput: 112.34349353995395	throughput: 52.86899251860919	throughput: 41.63865146451802
done	done	done	done	done

Comparison of 3 duplicate ack latency & throughput

timeout mode	timeout mode	timeout mode	timeout mode	timeout mode
timeout loss: 15 loss rate: 0.01	timeout loss: 44 loss rate: 0.05	timeout loss: 96 loss rate: 0.1	timeout loss: 135 loss rate: 0.15	timeout loss: 211 loss rate: 0.2
latency: 0.012378349884833204	latency: 0.039427899360656736	latency: 0.08066131925582885	latency: 0.09886782169342041	latency: 0.1511297194957733
throughput: 88.83845722833848	throughput: 25.362751153764314	throughput: 12.297516048905146	throughput: 10.114514337148992	throughput: 6.616832237473764
done	done	done	done	done

Comparison of timeout latency & throughput

It can clearly be seen that fast retransmission has much higher throughput than using timeout only. Since it is clear that a packet is lost when multiple duplicate acks are being received, it is reasonable to retransmit when 3 duplicate acks are detected. Throughput decreases dramatically from loss rate 1% to 5%. We can also observe that throughput and loss rate is inversely proportional.